Local Dependent Components

Arto Klami ARTO.KLAMI@TKK.FI
Samuel Kaski SAMUEL.KASKI@TKK.FI
Helsinki Institute for Information Technology and Adaptive Informatics Research Centre, Helsinki University of Technology, P.O. Box 5400, FI-02015 TKK, Finland

Abstract

We introduce a mixture of probabilistic canonical correlation analyzers model for analyzing local correlations, or more generally mutual statistical dependencies, in co-occurring data pairs. The model extends the traditional canonical correlation analysis and its probabilistic interpretation in three main ways. First, a full Bayesian treatment enables analysis of small samples (large p, small n, a crucial problem in bioinformatics, for instance), and rigorous estimation of the degree of dependency and independency. Secondly, the mixture formulation generalizes the method from global linearity to the more reasonable assumption of different kinds of dependencies for different kinds of data. As a third novel extension the method decomposes the variation in the data into shared and data set-specific components.

1. Introduction

We study the general framework of modeling or detecting dependencies between two (or in general more) data sets of co-occurring paired samples $(\mathbf{x}, \mathbf{y})$. Looking for statistical dependencies or commonalities between two (or in general more) measurements can be motivated from two different but closely related viewpoints. The first stems from noise reduction. If we have several measurements from noisy sensors that measure different properties of the same objects, combining the measurements in a suitable way reduces the amount of noise, and already naive approaches such as taking the average of two measurements of identical structure are often helpful. If noise is assumed to be independent between the sources then finding the dependencies is a principled way to reduce it. The sensors can be either real sensors such as gene expression arrays measuring the genome-wide expression levels, or "artificial sensors" such as text written in a specific language (see Li and Shawe-Taylor (2006) for such an approach). Regardless of the more specific setting it is often of interest to find what the measurements have in common.

The second source of motivation comes from analyzing what is interesting in the data. We can have measurements of very different types, each conveying different kind of information about the objects under consideration. For example, in an image search application we can have textual descriptions of images in addition to the actual pictorial content (Farquhar et al., 2006), or in a bioinformatics application we can have copy number aberration and expression measurements for the same genes (Berger et al., 2006). In such applications it is arguably a good idea to combine the different kinds of representations, because all measurements have been tailored to measure the same underlying phenomenon. This suggests that what is in common between them is really what we are interested in. Depending on the application we might then discard the variation that is specific to either data alone (i.e. consider it as noise, providing a direct link to the first source of motivation) or study separately what each of the data sets reveals in addition to the shared information.

The traditional approach to searching for dependencies is to assume some model family, select a dependency measure, and then optimize the model parameters to maximize the dependency. This leads to the classical method of canonical correlation analysis (CCA) (Hotelling, 1936) using linear projections as the model family and correlation as the dependency measure, and to various more recent methods that relax the linearity assumption or aim to maximize (an estimate of) the mutual information with different kind of parametric models (Dhillon et al., 2003; Friedman et al.,

Appearing in *Proceedings of the 24th International Conference on Machine Learning*, Corvallis, OR, 2007. Copyright 2007 by the author(s)/owner(s).

2001; Kaski et al., 2005; Verbeek et al., 2004). Also several kernel-based approaches have been presented, including kernelized CCA (see Shawe-Taylor and Cristianini (2004) for a textbook account). Kernel CCA with non-linear kernels discards the linearity assumption, but the possibility to interpret the dependencies in terms of the original dimensions is lost.

Here we take the opposite approach, and try to find dependencies with probabilistic generative models. Bach and Jordan (2005) interpreted CCA in a probabilistic way, offering a novel view on searching for dependencies. We extend the treatment to a Bayesian version of CCA, and present additional extensions to mixture models and simultaneously finding both the dependencies and the data set-specific variation as separate components. Using the mixture formulation we can look for dependencies in cases where it is not feasible to assume global linear dependency, whereas explicitly decomposing the variation may help in interpretation.

A heuristic mixture of CCAs has previously been presented by Fern et al. (2005), relying on pre-clustering the data points and applying traditional CCA for each cluster, and a non-Bayesian probabilistic mixture was briefly mentioned by Fyfe and Leen (2006). We treat the mixture case in length, and present an algorithm for sampling from the posterior distribution of the model. The performance of the sampler and properties of the model are then verified on generated toy data, and a demonstration of a practical application in bioinformatics is presented.

2. Canonical Correlation Analysis

In this work we study the classical CCA model which is currently receiving a lot of attention within the machine learning community (Archambeau et al., 2006; Bach & Jordan, 2005; Fyfe & Leen, 2006; Klami & Kaski, 2006; Leen & Fyfe, 2006). CCA is a prototype method for analyzing mutual dependency between data sets; it assumes linear projections which make it easily interpretable and keep the overfitting problems almost manageable. We identify some key directions of extensions, which are expected to generalize to other current extensions such as the robust variant of Archambeau et al. (2006).

The CCA can be computed by solving the generalized eigenvalue problem

$$\begin{pmatrix} 0 & \mathbf{\Sigma}_{xy} \\ \mathbf{\Sigma}_{yx} & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u}_x \\ \mathbf{u}_y \end{pmatrix} = \rho \begin{pmatrix} \mathbf{\Sigma}_{xx} & 0 \\ 0 & \mathbf{\Sigma}_{yy} \end{pmatrix} \begin{pmatrix} \mathbf{u}_x \\ \mathbf{u}_y \end{pmatrix},$$

where the $\mathbf{\Sigma}$ denote the covariance matrices, $\mathbf{\Sigma}_{xy}$ being between $\mathbf{x}$ and $\mathbf{y}$ etc. The eigenvalues ρ are the canonical correlations, and the eigenvectors $\mathbf{u}$ contain the canonical weights. Perhaps a bit more intuitive way to think about CCA is to look at it just as a method maximizing the correlation between the projections $\mathbf{u}_x^T\mathbf{x}$ and $\mathbf{u}_y^T\mathbf{y}$, often called canonical scores, with suitable restrictions on the projections.

2.1. Probabilistic CCA

Much of the recent work on CCA has been based on the probabilistic interpretation by Bach and Jordan (2005), which is briefly summarized here. For the remainder of the paper the probabilistic model is abbreviated as PCCA, while CCA refers to the classical CCA method.

Denote by $\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_n]$ and $\mathbf{Y} = [\mathbf{y}_1, ..., \mathbf{y}_n]$ the data sets of paired samples. PCCA is parameterized by $\boldsymbol{\theta} = \{\mathbf{W}, \mathbf{\Psi}_x, \mathbf{\Psi}_y, \boldsymbol{\mu}, \mathbf{Z}\}$, where $\mathbf{Z} = [\mathbf{z}_1, ..., \mathbf{z}_n]$ are latent variables, one for each sample, and the rest are model parameters. The model can be written as

$$\begin{aligned} \mathbf{z}_j &\sim N(0, \mathbf{I}), \\ (\mathbf{x}_j, \mathbf{y}_j) &\sim N(\boldsymbol{\mu} + \mathbf{W}\mathbf{z}_j, \mathbf{\Psi}), \end{aligned} \tag{1}$$

where $\mathbf{\Psi}$ is a block-diagonal matrix that has $\mathbf{\Psi}_x$ and $\mathbf{\Psi}_y$ on its diagonal. Alternatively, we could write the likelihood for both data spaces separately, yielding $\mathbf{x}_j \sim N(\boldsymbol{\mu}_x + \mathbf{W}_x\mathbf{z}_j, \mathbf{\Psi}_x)$ where $\boldsymbol{\mu}_x$ and $\mathbf{W}_x$ are the parts of $\boldsymbol{\mu}$ and $\mathbf{W}$ that correspond to $\mathbf{x}$, and likewise for $\mathbf{y}$. The crucial thing is that the latent variables $\mathbf{z}$ are shared between the two data sets, while everything else is independent.

Bach and Jordan (2005) showed the connection to CCA by proving that in the maximum likelihood solution of (1) we have $\mathbf{W}_x = \mathbf{\Sigma}_{xx}\mathbf{U}_x\mathbf{P}^{1/2}\mathbf{R}$ and $\mathbf{W}_y = \mathbf{\Sigma}_{yy}\mathbf{U}_y\mathbf{P}^{1/2}\mathbf{R}$, where the $\mathbf{U}$ contain the canonical weights, $\mathbf{P}$ is a diagonal matrix containing the canonical correlations, and $\mathbf{R}$ is an arbitrary rotation matrix. Furthermore, the expectations of the latent variables, $E[\mathbf{z}|\mathbf{x}]$ and $E[\mathbf{z}|\mathbf{y}]$, lie in the subspace that the traditional CCA finds. The maximum likelihood solution can be found by an EM algorithm, guaranteed to converge to the global optimum but leaving the arbitrary rotation $\mathbf{R}$ undetermined. The rotational ambiguity can be solved in a straightforward way (Archambeau et al., 2006), however, giving the individual components.[1]

3. Bayesian CCA

Classical CCA is known to overfit badly to small data sets, detecting artificially high correlations. This can

[1] The Appendix explaining the procedure is incorrect in the original publication, but is corrected in an errata.

be especially harmful in exploratory data analysis aiming to form hypotheses of data; spurious correlations lead to clearly incorrect hypotheses and wasted effort in further studies. Moreover, it is difficult in practice to identify how many canonical correlation components there are, even though some statistical tests have been proposed. The probabilistic interpretation as such does not solve these problems, but it gives a starting point for a Bayesian model which has the necessary tools.

The probabilistic model in (1) is here extended to a Bayesian generative model (BCCA) by introducing suitable prior distributions, in particular to tackle the issue of detecting independencies, and by providing a method for inference. In this paper we use Gibbs sampling, but variational approximation would be feasible as well.

The likelihood and the prior for the latent variables are exactly as in (1), and the prior distributions for the model parameters are the following:

$$\begin{aligned} \mathbf{w}_i &\sim N(0, \beta_i \mathbf{I}), \\ \beta_i &\sim IG(\alpha_0, \beta_0), \\ \mathbf{\Psi}_x, \mathbf{\Psi}_y &\sim IW(\mathbf{S}_0, \nu_0), \\ \boldsymbol{\mu} &\sim N(0, \sigma^2 \mathbf{I}). \end{aligned} \tag{2}$$

Here $\mathbf{w}_i$ denotes the ith column of $\mathbf{W}$, and IG and IW are shorthand notations for the inverse Gamma and inverse Wishart distributions. The priors for the mean $\boldsymbol{\mu}$ and the covariance matrices $\mathbf{\Psi}_x$ and $\mathbf{\Psi}_y$ are conventional conjugate priors, and the prior for the projection matrices is the so-called Automatic Relevance Determination (ARD) prior used for example in Bayesian principal component analysis (Bishop, 1999). We use values $\alpha_0 = 0.1$, $\beta_0 = 0.1$, $\nu_0 = p+1$, where p is the dimensionality of $\mathbf{x}$ or $\mathbf{y}$, $\mathbf{S}_0 = \mathbf{I}$, and $\sigma^2 = 1$ in all experiments to provide reasonably vague and generally applicable priors.

The purpose of the ARD prior here is to automatically control the number of components extracted by the model. The parameter β_i controls the magnitude of $\mathbf{w}_i$: If the dimensionality of the dependent subspace is less than the full dimensionality of $\mathbf{W}$ the prior variance parameter for the remaining columns goes towards zero, as do the actual elements of the vectors.

A Gibbs sampler will be used to draw samples from the posterior distribution. The conditional probability distribution of each variable given the rest has a form of some classic distribution and thus sampling is straightforward. The formulas are

$$\begin{aligned} \beta_i|\mathbf{w}_i &\sim IG(\frac{1}{2}p_{\mathbf{v}} + \alpha_0, \frac{1}{2}\mathbf{w}_i^T\mathbf{w}_i + \beta_0), \\ \mathbf{\Psi}_x|\tilde{\mathbf{X}}, \mathbf{W}, &\sim IW(\mathbf{S}_0 + \mathbf{S}, \nu_0 + n), \\ \text{where } \mathbf{S} &= \sum_j (\tilde{\mathbf{x}}_j - \mathbf{W}\mathbf{z}_j)(\tilde{\mathbf{x}}_j - \mathbf{W}\mathbf{z}_j)^T, \\ \boldsymbol{\mu}|\mathbf{V}, \mathbf{\Psi}, \mathbf{W} &\sim N(\mathbf{\Sigma}\mathbf{\Psi}^{-1}\sum_j \mathbf{v}_j, \mathbf{\Sigma}), \\ \text{where } \mathbf{\Sigma}^{-1} &= n\left(\mathbf{W}\mathbf{W}^T + \mathbf{\Psi}\right)^{-1} + \frac{1}{\sigma^2}\mathbf{I}, \\ \mathbf{w}_i|A &\sim N(\mathbf{\Psi}^{-1}\mathbf{\Sigma}\sum_j z_{ij}(\tilde{\mathbf{v}}_j - \mathbf{w}_{-i}z_{-ij}), \mathbf{\Sigma}), \\ \text{where } \mathbf{\Sigma}^{-1} &= \sum_j z_{ij}^2 \mathbf{\Psi}^{-1} + \frac{1}{\beta_i}\mathbf{I} \;\text{ and} \\ \mathbf{z}_j|\tilde{\mathbf{v}}_j, \mathbf{W}, \mathbf{\Psi} &\sim N(\mathbf{W}^T\mathbf{\Sigma}\tilde{\mathbf{v}}_j, \mathbf{I} - \mathbf{W}^T\mathbf{\Sigma}\mathbf{W}), \\ \text{where } \mathbf{\Sigma}^{-1} &= \mathbf{W}\mathbf{W}^T + \mathbf{\Psi}. \end{aligned} \tag{3}$$

The shorthand notation A in the sampling formula of $\mathbf{w}_i$ denotes $\tilde{\mathbf{V}}, \mathbf{w}_{-i}, \mathbf{Z}, \mathbf{\Psi}$, and β_i. The notation z_{ij} refers to the element on ith row and jth column in $\mathbf{Z}$ (or equivalently the ith element of $\mathbf{z}_j$), and negative indices mean every column/row except the one mentioned. The variable $\mathbf{V}$ is the row-wise concatenation of $\mathbf{X}$ and $\mathbf{Y}$, $\tilde{\mathbf{v}}_j$ means $\mathbf{v}_j - \boldsymbol{\mu}$, $p_{\mathbf{v}}$ is the number of dimensions in $\mathbf{v}$, and n is the number of samples. In all formulas $\mathbf{\Psi} = [\mathbf{\Psi}_x, 0; 0, \mathbf{\Psi}_y]$, a block-diagonal matrix with $\mathbf{\Psi}_x$ and $\mathbf{\Psi}_y$ on its diagonal. The sampling formula for $\mathbf{\Psi}_y$ is identical to that of $\mathbf{\Psi}_x$, but naturally using $\mathbf{y}$ instead of $\mathbf{x}$.

The only somewhat more complicated part is the sampling of the projection matrices $\mathbf{W}$. The prior for the matrix as a whole is not conjugate, so we sample the values component by component (i.e., one column of $\mathbf{W}$ at a time). As the order of the components is arbitrary the columns of $\mathbf{W}$ (and correspondingly the rows of $\mathbf{Z}$ and $\boldsymbol{\beta}$) will be re-arranged after each iteration according to their magnitude to improve convergence.

The sampler converges to the posterior distribution of the model parameters, after which inference on model parameters can be done using the posterior samples. The convergence is here measured by the potential scale reduction factor by Brooks and Gelman (1998), applied to the total likelihood of the model given the data $\{\mathbf{x}_j, \mathbf{y}_j\}_{j=1}^n$. The parameter values could alternatively be used for stricter convergence control.

4. Local Dependent Components

To make modeling of dependencies local and to get rid of the assumption of global linear dependency we introduce a mixture of Bayesian canonical correlation analyzers. The model is formulated as a Dirichlet process mixture to avoid having to choose the number of clusters. The process is described by

$$\begin{aligned}\boldsymbol{\phi}_j &\sim G \\ G &\sim DP(G_0, \alpha),\end{aligned}$$

where G_0 is the joint prior distribution specified in (2), $\boldsymbol{\phi}_j$ includes $\mathbf{W}, \boldsymbol{\Psi}, \boldsymbol{\beta}$ and $\boldsymbol{\mu}$ for the jth sample, and DP denotes the Dirichlet process with α as a concentration parameter. Note that while $\boldsymbol{\phi}_j$ is drawn separately for each data point, the Dirichlet process has a clustering effect by giving identical values for several points. Here G_0 is not conjugate to the model, but only conditionally conjugate in the sense that the joint prior can be written as a product of conjugate priors. This means that we are not able to integrate out the model parameters, and thus cannot use the conventional approaches, such as (MacEachern, 1994), for sampling from the Dirichlet process mixture.

We can, however, draw posterior samples using a non-conjugate split-merge procedure (Jain & Neal, 2005). The procedure works by suggesting to either split an existing cluster in two, or to merge two clusters into one. When splitting, the algorithm draws two new components from the prior and uses restricted Gibbs sampling to approach reasonable states for the components. Note that since we sample from the prior, having somewhat informative prior improves convergence. The restricted Gibbs here means a sampler which only considers the two components and the collection of samples assigned to them, and thus does not vary the number of components. After some restricted scans a final Gibbs step is performed, and the proposal is either accepted or rejected based on the Metropolis-Hastings ratio. The merging proceeds similarly, also involving restricted Gibbs sampling for an imaginary split. The split-merge procedure is accompanied by incremental Metropolis-Hastings updates (algorithm 5 in (Neal, 1998)) as suggested in Jain and Neal (2005).

The sampling formulas required in these algorithms are the ones given in (3), this time always conditioned on the samples in a given cluster, together with a new distribution for sampling $\mathbf{s}$, a latent vector indicating the cluster membership of each sample. The restricted Gibbs steps use

$$\mathbf{s}_j|\mathbf{v}_j, \mathbf{W}, \boldsymbol{\Psi} \propto \frac{n_{-j,k}+\alpha}{n-1+\alpha} N(\mathbf{v}_j|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$
$$\text{where } \boldsymbol{\Sigma}_k = \mathbf{W}_k\mathbf{W}_k^T + \boldsymbol{\Psi}_\mathbf{k},$$

and $n_{-j,k}$ denotes the number of samples in the kth cluster, excluding the jth sample itself. Note the slight notational abuse as the subscript in $\mathbf{W}, \boldsymbol{\mu}$, and $\boldsymbol{\Psi}$ now refers to the cluster instead of elements, and $N(\mathbf{v}_j|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the probability density of the given normal distribution at $\mathbf{v}_j$. The incremental Metropolis-Hastings algorithm samples directly from the prior as

$$\mathbf{s}_j = \frac{n_{-j,k}}{n-1+\alpha},$$

together with an additional probability $\alpha/(n-1+\alpha)$ for creating a new component, and accepts or discards these samples based on the normal density. We consider α here as being fixed, so no update is needed for it, and used $\alpha = 1$ in all experiments.

5. Decomposition of Variation

In data analysis it may be interesting to know both what is specific to each data set and what is shared, instead of only searching for the shared aspects or dependencies as CCA does. In the model (1) the data set-specific variation is modeled only implicitly by the covariance matrix parameters $\boldsymbol{\Psi}$. To make this variation interpretable we decompose it into components as well. This whole procedure results in a decomposition of all variation into three sets of components: The first models the shared variation, the second variation specific to $\mathbf{x}$, and the third variation specific to $\mathbf{y}$.

To achieve this we use a method that is effectively Bayesian PCA (Bishop, 1999), implemented with sampling. For given values of the BCCA model parameters, it is straightforward to factorize the covariance matrices $\boldsymbol{\Psi}_x$ and $\boldsymbol{\Psi}_y$ individually, by sampling from the posterior of the model

$$\mathbf{x}_j \sim N(\mathbf{B}_x\mathbf{t}_j, \boldsymbol{\Sigma}_x + \sigma_x^2\mathbf{I}),$$

where $\mathbf{t}_j$ is latent variable for the sample $\mathbf{x}_j$, $\mathbf{B}_x$ contains the projection vectors, and $\boldsymbol{\Sigma}_x = \mathbf{W}_x\mathbf{W}_x^T$ corresponds to the variation already explained by the BCCA model. The priors and sampling formulas for $\mathbf{B}$ and $\mathbf{t}$ are similar to (2) and (3); details are omitted here for brevity. For σ_x^2 an inverse Gamma prior is used, and it is sampled with Metropolis-Hastings in the logarithmic domain.

To obtain the factorization for all posterior samples of BCCA without needing to run a long sampling chain for each, we use two interleaved MCMC-chains. One updates the parameters of the BCCA model and works exactly as in (3), not taking the data set-specific factorizations into account at all. The other chain then samples a few samples from the factorization chains, given the most recent BCCA sample and starting from the sample obtained using the previous one. When we cannot assume that two consecutive BCCA samples are close enough, in practice after accepting a split or merge proposal in the mixture model, the factorization chains are sampled longer to ensure convergence to the new BCCA model.

An alternative way to do the same would be to extend the model to explicitly model data set-specific variation together with the dependencies as suggested in (Klami & Kaski, 2006). However, as shown in that paper PCCA (and consequently BCCA) works correctly only if the data set-specific variation is exactly marginalized out, and thus directly sampling from this kind of a model could be difficult.

6. Experiments

In this section we demonstrate and experimentally verify properties of the model. First it is shown that when applied to data sets having small n, BCCA is able to avoid overlearning, in contrast to classical CCA. Next, the decomposition into shared and data set-specific variation is demonstrated on toy data. Third, the ability of the split-merge sampler to detect local dependencies is verified using a clustered data set. Finally, we present a brief example of how the mixture model could be applied on real biological data.

6.1. Performance on Small Data Sets

We first demonstrate that BCCA performs equally to the classical CCA on data sets with a large number of samples, and then show that it is considerably less prone to overfitting when applied to smaller data sets.

We created two four-dimensional data sources having jointly Gaussian distribution with three correlating dimensions (correlations 0.7, 0.3 and 0.1), and compared the methods on large sets ($n = 800$) and small sets ($n = 50$). For classical CCA the uncertainty in the estimates was evaluated using non-parametric bootstrapping.

Given a sufficiently large data set the results of Bayesian CCA and normal CCA are effectively equal (Figure 1). Both methods find the true correlations, and the widths and shapes of the distribution of correlations are also comparable. Traditional CCA, however, occasionally finds spurious correlations even from a data set this large.

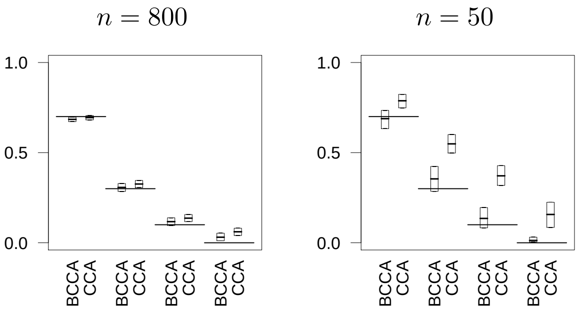

Figure 1. Boxplots of the correlations extracted by Bayesian CCA and normal CCA. The real correlations of the four components are marked with horizontal lines, and results for two different data set sizes are reported. On the larger data set (left) the results of the two methods are very similar, but for the smaller data set (right) classical CCA overfits seriously, in particular for the smaller correlations. For BCCA the most notable difference between the data set sizes is in the width of the posterior distributions. The boxes extend to the 25% and 75% quantiles and the tick marks the median.

With smaller data sets normal CCA overfits by giving artificially high correlations. Especially the smaller correlations are considerably overestimated. The Bayesian variant fares a lot better, mostly showing just wider posterior distributions compared to the larger data and capturing the true correlation in between the 25% and 75% quantiles in every case.

While $n = 50$ may sound like a very small number of samples, it is worth remembering that both data sets had just $p = 4$ dimensions. In many biological data sets the ratio is even worse, to the degree that we may even have $n \ll p$ if considering for example genes as features and patients or conditions as samples.

6.2. Decomposition of Variation

Here we demonstrate how the variation can be decomposed into shared and data set-specific components, on a simple jointly Gaussian data set with a manually specified covariance matrix. The data has one common (shared) component and one component specific to each data set. The obtained projections are compared to two naive alternatives that do not aim at such a decomposition: principal component analysis (PCA) of each individual data set separately, and PCA of the concatenated data $[\mathbf{x}; \mathbf{y}]$. The latter is a global linear model of the whole data collection.

The first BCCA projection vectors (means over the posterior samples) for shared and data set-specific variation are presented in Figure 2. For comparison we have included three first components of the PCA of the concatenated data to illustrate that while a standard joint model describes the data well it makes no distinction between what is shared and what is specific to each data source. Parts of both are included in the same components. The data set-specific PCAs naturally could not distinguish those in any way.

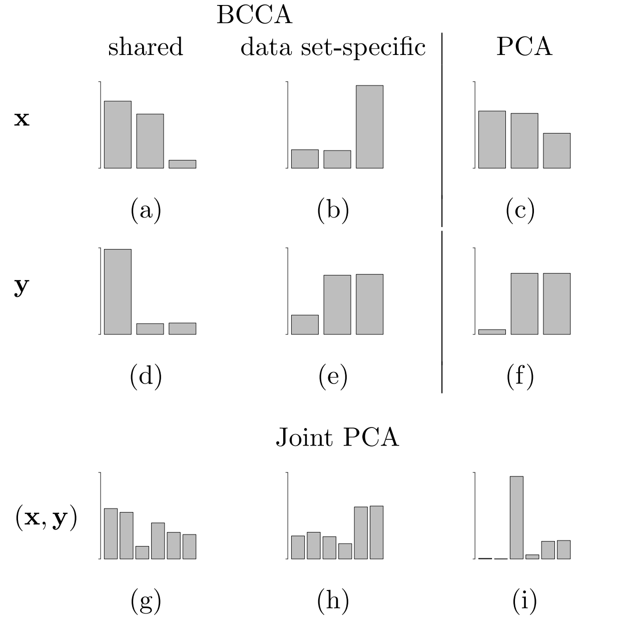

Figure 2. Illustration of a decomposition of the variation into shared and data set-specific components. The bottom line (g-i) displays the first three projection vectors of a PCA applied to the concatenation of $\mathbf{x}$ and $\mathbf{y}$. This kind of joint modeling of the whole data collection suggests that the most dominating components (g-h) involve all dimensions with relatively high weights. The BCCA reveals that actually the joint analysis mixed two separate effects: a shared component involving the first two dimensions of $\mathbf{x}$ and the first dimension of $\mathbf{y}$ (a,d), and the data set-specific variations (b,e). The third joint component (i) corresponds to variation that is almost solely specific to $\mathbf{x}$. PCA of each data set alone (c,f) naturally cannot distinguish between shared and data set-specific variation, and simply finds the main source of variation in each source. The bars represent the absolute values of the projection weights, indicating the importance of the dimensions. The scale is from zero to one in every subfigure.

6.3. Local Dependent Components

The mixture variant for performing local analysis is demonstrated on data with clear but still overlapping clusters. This illustrates that the somewhat complicated split-merge procedure works in practice, finding the clusters and the dependencies within each.

Table 1. Correlations found by a mixture of BCCAs. The reported ranges are the 25% and 75% quantiles of the posterior distribution, and we see that in most cases the larger correlations are well matched. The smaller ones are somewhat overestimated, in particular for the clusters 2 and 3 having only a two-dimensional dependent subspace.

Cluster 1		Cluster 2		Cluster 3	
true	BCCA	true	BCCA	true	BCCA
0.7	0.68-0.71	0.5	0.50-0.55	0.9	0.85-0.86
0.3	0.28-0.34	0.2	0.19-0.25	0.6	0.51-0.56
0.1	0.12-0.18	0	0.03-0.07	0	0.05-0.09
0	0.01-0.05	0	0.00-0.01	0	0.00-0.02

For this purpose we generated a total of 1500 samples from three slightly overlapping (the average distance between centroids was twice the average standard deviation of dimensions) jointly Gaussian clusters having an equal size but varying degree of manually specified correlations. The number of detected clusters was between 3 and 7 in all posterior samples, and while the number was usually (in 82% of samples) above 3, the extra clusters mostly contained only a few data points. On average 1450 of the 1500 data points were in the three biggest clusters, demonstrating that the sampler was able to find the true cluster structure. The posterior intervals of the correlations within each true cluster are reported in Table 1. These figures were collected after matching, for each posterior sample, the clusters of the model to the true clusters by majority voting. In practical data analysis this label switching problem could be solved by an additional standard clustering step as is done in the next section or, for instance, by computing a variational approximation. The true correlations are found relatively well, although the method does find some weak spurious correlations.

Note that traditional CCA (or a single BCCA) would here provide completely different results as the projections would try to capture the clustering effect in addition to the within-cluster dependencies.

6.4. Regulation of Heat Shock in Yeast

We next apply the mixture model to analyzing the regulation of heat shock in *Saccharomyces cerevisiae*, as a demonstration of potential applications. The task is to find groups of genes that share effective regulators and are regulated differently from the rest of the genes in the specified environmental conditions. The information is extracted from two data sets, one being a time series (8 time points) of gene expression in heat shock (Gasch et al., 2000), and the other ChIP chip measurements of how well 6 different transcription factors bind into the promoter region of each gene in yeast grown under heat shock (Harbison et al., 2004).

Gene regulation is typically modeled by Bayes networks, which have the problem that structure search is very difficult for small data sets. In practice the set of genes would need to be restricted, and we could in principle first cluster the genes into smaller modules, and then search for regulatory relationships within each by building a bi-partite graphical model to represent the dependencies. Links between the two parts, transcription factor binding and a time point in the expression data, would tell about the dependencies, whereas links within each part would explain the data set-specific variation. The mixture of CCAs is an alternative exploratory method that performs both the clustering and extraction of the dependency structure in one rigorous step, represents the shared and data set-specific variation in the form of easily interpretable latent variables (projections on linear components), and most importantly is generally usable in other kinds of applications as well.

We applied the split-merge sampler to a random subset of 2000 genes. From the collection of the posterior samples it is possible to compute summary statistics about dependency either globally or locally for each gene or transcription factor or pairs of them, or construct lists of similarly regulated genes or potential regulators. For instance, similarity of regulation of genes can be quantified (as the likelihood of belonging to the same mixture component), and effectiveness of co-binding of transcription factors can be assessed.

Here we summarize main findings for one consistent cluster. The most consistent clusters were extracted using a complete linkage hierarchical clustering algorithm on the matrix of pairwise probabilities for two genes to belong to the same cluster. This provided four reasonably consistent clusters, each showing different kinds of dependencies. Some main observations for one of the clusters are presented in Figure 3.

7. Discussion

In this paper a Bayesian version of probabilistic canonical correlation analysis and a sampling procedure for the model were presented to address two problems of traditional CCA: overfitting for small data sets, and the need for assuming global linear dependency. Furthermore, an extension that is able to extract also data set-specific variation in addition to the dependencies was presented. All of these properties were verified on generated data, and an example application of the method on biological data was presented.

The sampling approach used in this paper does not require coarse approximations, but for the mixture model it is very time-consuming and introduces a "label switching problem": it is non-trivial to identify clusters in several different posterior samples. While there are approximate methods for solving the problem, and global and data point-specific statistics on dependency are more than sufficient in many applications, it is worthwhile to complement the method by point estimates. For example variational Bayes could be useful for practical applications and is definitely worth considering. A single Bayesian CCA model, however, is effectively computable for real applications already with the current sampling method.

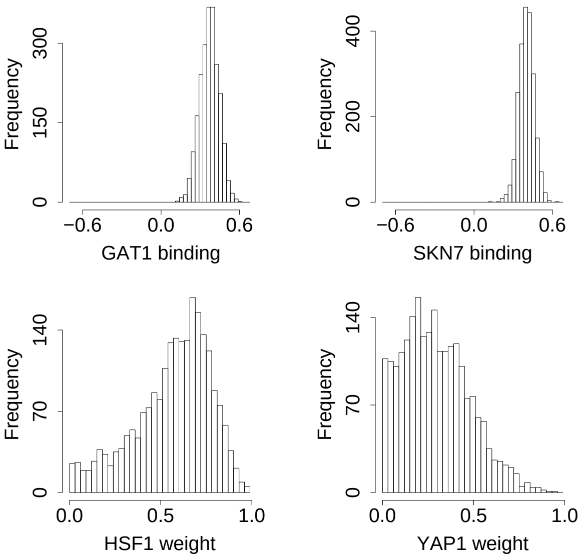

Figure 3. Sample regulation effects found in yeast heat shock by the Bayesian mixture of CCAs, presented for one cluster. The histograms of the posterior samples of the mean binding levels for two transcription factors, GAT1 and SKN7, are shown on the top row. Values above zero indicate higher than average binding, and thus the histograms show that GAT1 and SKN7 bind to the promoter regions of the genes in this cluster during stress. The bottom row displays the posterior distribution of the weights of the first CCA projection vector corresponding to two transcription factors, HSF1 and YAP1. The high values indicate that the binding of these two factors correlates with the expression of the genes. Taken together, this information suggests that HSF1 and YAP1 are important regulators for a group of genes that are already bound by GAT1 and SKN7. SKN7 is known to form a two-component signaling system in oxidative stress with both HSF1 and YAP1, providing support for the finding (see `http://www.yeastgenome.org/`).

Acknowledgments

This work was supported by the Academy of Finland, decision number 207467, and in part by the IST Programme of the European Community, under the PASCAL Network of Excellence, IST-2002-506778. This publication only reflects the authors' views. All rights are reserved because of other commitments.

References

Archambeau, C., Delannay, N., & Verleysen, M. (2006). Robust probabilistic projections. *Proceedings of the 23rd International Conference on Machine Learning* (pp. 33–40). New York, NY: ACM Press.

Bach, F. R., & Jordan, M. I. (2005). *A probabilistic interpretation of canonical correlation analysis* (Technical Report 688). Department of Statistics, University of California, Berkeley.

Berger, J. A., Hautaniemi, S., Mitra, S. K., & Astola, J. (2006). Jointly analyzing gene expression and copy number data in breast cancer using data reduction models. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, *3*, 2–16.

Bishop, C. M. (1999). Bayesian PCA. *Advances in Neural Information Processing Systems 11* (pp. 382–388). Cambridge, MA: MIT Press.

Brooks, S., & Gelman, A. (1998). General methods for monitoring convergence of iterative simulations. *Journal of Computational and Graphical Statistics*, *7*, 434–456.

Dhillon, I. S., Mallela, S., & Modha, D. S. (2003). Information-theoretic co-clustering. *Proceedings of KDD'03, the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 89–98). New York, NY: ACM Press.

Farquhar, J. D. R., Hardoon, D. R., Meng, H., Shawe-Taylor, J., & Szedmak, S. (2006). Two view learning: SVM-2K, theory and practice. *Advances in Neural Information Processing Systems 18* (pp. 355–362). Cambridge, MA: MIT Press.

Fern, X., Brodley, C. E., & Friedl, M. A. (2005). Correlation clustering for learning mixtures of canonical correlation models. *Proceedings of the Fifth SIAM International Conference on Data Mining* (pp. 439–448).

Friedman, N., Mosenzon, O., Slonim, N., & Tishby, N. (2001). Multivariate information bottleneck. *Proceedings of UAI'01, the 17th Conference on Uncertainty in Artificial Intelligence* (pp. 152–161). San Francisco, CA: Morgan Kaufmann Publishers.

Fyfe, C., & Leen, G. (2006). Stochastic processes for canonical correlation analysis. *Proceedings of the 14th European Symposium on Artificial Neural Networks* (pp. 245–250).

Gasch, A. P. et al. (2000). Genomic expression programs in the response of yeast cells to environmental changes. *Molecular Biology of the Cell*, *11*, 4241–4257.

Harbison, C. et al. (2004) Transcriptional regulatory code of a eukaryotic genome. *Nature*, *431*, 99–104.

Hotelling, H. (1936). Relations between two sets of variates. *Biometrika*, *28*, 321–377.

Jain, S., & Neal, R. (2005). *Splitting and merging components of a non-conjugate Dirichlet process mixture model* (Technical Report 0507). Department of statistics, University of Toronto.

Kaski, S., Nikkilä, J., Sinkkonen, J., Lahti, L., Knuuttila, J., & Roos, C. (2005). Associative clustering for exploring dependencies between functional genomics data sets. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, *2*, 203–216.

Klami, A., & Kaski, S. (2006). Generative models that discover dependencies between data sets. *Machine Learning for Signal Processing XVI* (pp. 123–128). IEEE.

Leen, G., & Fyfe, C. (2006). A Gaussian process latent variable model formulation of canonical correlation analysis. *Proceedings of the 14th European Symposium on Artificial Neural Networks* (pp. 418–418).

Li, Y., & Shawe-Taylor, J. (2006). Using KCCA for Japanese-English cross-language information retrieval and document classification. *Journal of Intelligent Information Systems*, *27*, 117–133.

MacEachern, S. (1994). Estimating normal means with a conjugate style Dirichlet process prior. *Communications in Statistics: Simulation and Computation*, *23*, 727–741.

Neal, R. (1998). *Markov chain sampling methods for Dirichlet process mixture models* (Technical Report 9815). Department of statistics, University of Toronto.

Shawe-Taylor, J., & Cristianini, N. (2004). *Kernel methods for pattern analysis*. Cambridge: Cambridge university press.

Verbeek, J., Roweis, S., & Vlassis, N. (2004). Nonlinear CCA and PCA by alignment of local models. *Advances in Neural Information Processing Systems 16* (pp. 297–304). Cambridge, MA: MIT Press.